UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/547,290	04/11/2000	Nir N. Shavit	1004-4664	4986

22120 7590 04/21/2005

ZAGORIN O'BRIEN GRAHAM LLP
7600B N. CAPITAL OF TEXAS HWY.
SUITE 350
AUSTIN, TX 78731

EXAMINER	
BULLOCK JR, LEWIS ALEXANDER	
ART UNIT	**PAPER NUMBER**
2195	

DATE MAILED: 04/21/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/547,290	SHAVIT ET AL.	
	Examiner	Art Unit	
	Lewis A. Bullock, Jr.	2195	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *03 January 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-23* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☒ Claim(s) *12-22* is/are allowed.

6)☒ Claim(s) *1-8,11 and 23* is/are rejected.

7)☒ Claim(s) *9 and 10* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *17 April 2000* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Allowable Subject Matter

1. Claims 12-22 are allowed.

2. Claims 9 and 10 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

3. The following is a statement of reasons for the indication of allowable subject matter: The cited claims detail concurrent access is mediated on a linked-list structure by performing during execution of each of access operations, an atomic update of a respective one of the opposing-end indices and of an array element corresponding thereto. The cited prior art of record does not teach this atomic operation being performed on a linked list or enabling concurrent non-blocking access to the linked list. "A simple and correct queue algorithm using compare and swap" by Stone teaches an array structure wherein an enqueue or dequeue operation enables the changing of the head and tail identifiers with the linking of the node. "A Non-Blocking Algorithm for Concurrent Data Structures" and "A simple and correct shared queue algorithm using Compare and Swap" by Prakash both teaches the breaking up of this operation into two steps. WO/86/00434 publication teaches a linked list wherein access operations execute an atomic operation that changes a list-node end identifier and the markings of the node. However, the publication does not operate in a concurrent non-blocking environment and requires one to enable exclusive access to changing the list. Therefore, the cited claims are allowable over the prior art of record.

Claim Rejections - 35 USC § 102

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

2. Claims 1, 4, and 23 are rejected under 35 U.S.C. 102(b) as being anticipated by "A simple and correct shared queue algorithm using Compare and Swap" by STONE.

As to claim 1, STONE teaches a concurrent shared object representation (shared queue) comprising: a computer readable encoding for a sequence of zero or more values (shared object) (pg. 497, "We assume our collection of shared objects is realized as an array."); and access operations (enqueue / dequeue) defined for access to each of opposing ends of the sequence (head / tail) (pg. 498, "To append an item to the queue a process sets the tail pointer to the new item and then links the old tail item, assuming there was one to the new item."), wherein execution of any one of the access operations is non-blocking with respect to any other execution of the access operations throughout a complete range of valid states (pg. 499, "Since multiple enqueue operations can be occurring concurrently, and because enqueue takes two steps..."), including one or more boundary condition states (empty states) (pg. 503, "If there is an item in a queue or in the process of being enqueed then an item can be dequened within a finite amount of time....We have already observed that new enqueue and dequeue operations can begin at any time and will proceed to successful conclusion."),

and wherein at least for those of the valid states other than the one or more boundary condition states, opposing ends ones of the access operations are disjoint (pg. 499, "Try to set the tail pointer. Stay in the retry loop until the Compare-and-Swap succeeds. If it fails another processor succeeded in swapping the tail pointer to a different item...").

As to claim 4, STONE teaches wherein the access operations include push and pop operations (enqueue and dequeue) (pg. 498-499).

As to claim 23, STONE teaches an apparatus (system) comprising: plural processors (processors) (abstract); a store addressable by each of the plural processors (memory storing head and tail pointers); first and second end identifier stores (memory storing copy of head and tail pointers) accessible to each of the plural processors for identifying opposing ends of a concurrent shared object in the addressable store (pg. 499, "Make a private copy "PrivateT", of the tail pointer."; "Make a private copy, PrivateH, of the head pointer."); and means for coordinating competing pop operations (dequeue operations), the coordinating means employing in each instance thereof, an atomic operation (CSDBL) to disambiguate a retry state and a boundary condition state of the concurrent shared object based on then-current contents of one of the first and second end-identifier stores (head / tail pointers) and an element of the concurrent shared object (NIL) corresponding thereto (determination that the queue is empty) (pg. 503, This queue algorithm is nondelaying: it is never necessary for one processor to wait for another processor to do something....It is fair in the sense

that processors contending for Compare-and-Swap access to a shared pointer having equal probability of success. However, starvation is possible: it is possible for a processor to fail all its attempts to modify a shared pointer. Our liveness properties say that queue operations can always proceed successfully").

3. Claims 1 and 3-5 are rejected under 35 U.S.C. 102(b) as being anticipated by "A Nonblocking Algorithm for Shared Queues Using Compare and Swap" by PRAKASH.

As to claim 1, PRAKASH teaches a concurrent shared object representation (linked-list) comprising: a computer readable encoding for a sequence of zero or more values (objects) (pg. 549" The data structure we use for the shared FIFO queue is a singly linked list."); and access operations (enqueue / dequeue) defined for access to each of opposing ends of the sequence (head / tail) (pg. 549, "Objects are dequeued at the head and enqueued at the tail of the linked list."), wherein execution of any one of the access operations is non-blocking with respect to any other execution of the access operations throughout a complete range of valid states, including one or more boundary condition states (empty states) (state 7) (pg. 550, "An enqueuer can proceed with its operation from only three states; "A dequeuer can proceed with its operation only from states 1,2, 3, and 7."), and wherein at least for those of the valid states other than the one or more boundary condition states, opposing ends ones of the access operations are disjoint (pg. 551, "If the head and the tail of the queue are different, serial changes may be made simultaneously at the head and at the tail...If there are enqueue and dequeue attempts being made on the queue, some attempt will succeed in a finte amount of time. The liveness property ensures that some enqueuer or dequeuer must

succeed in performing its operation in a finite amount of time regardless of the failure or inactivity of other enquirers and dequeuers. Since the liveness property is unconditional, this algorithm is nonblocking.").

As to claim 3, PRAKASH teaches the computer readable encoding includes a linked-list of nodes (linked-list of objects) representing the sequence; and wherein the one or more boundary condition states include one or more empty states (empty) (state 7) (pg 549 and 550).

As to claim 4, PRAKASH teaches the access operations include push and pop operations (enqueue and dequeue) (pg. 550-551).

As to claim 5, PRAKASH teaches the access operations include delete operations (pg. 550, "The object is marked for deletion by changing mark of the nextobject pointer from ENQ to DEQ.").

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 2 and 6-8 are rejected under 35 U.S.C. 103(a) as being unpatentable over "A simple and Correct Shared-Queue Algorithm using Compare-and-Swap" by STONE.

As to claim 2, STONE teaches the computer-readable encoding includes an array of elements for representing the sequence (pg. 497, "We assume our collection of shared objects is realized as an array."); and the one or more boundary condition states include a full state an empty state (empty state). Official Notice is taken in that it is well known in the art that a queue also has a full state. Therefore, it would be obvious to one skilled in the art that determine whether the queue is empty of full.

As to claim 8, STONE teaches the array of elements (array) with opposing-end indices (head / tail) respectively identifying opposing ends of the sequence (array); and wherein concurrent non-blocking access is mediated, at least in part, by performing during execution of each of the access operations (enqueue / dequeue), an atomic update (via CSDBL) of a respective one of the opposing-end indices (head / tail) and of an array element corresponding thereto (pg. 499, "When the Compare and Swap succeeds, Qtail points to the new tail and PrivatT points to the new tail.; pg. 498, To append an item to the queue, a process sets the tail pointer to the new item and then links the old tail item, assuming there was one, to the new item."). Official Notice is taken in that it is well known in the art that a array is constructed as a circular buffer of fixed size and therefore obvious that the array of STONE is a circular buffer.

As to claims 6 and 7, STONE teaches wherein the access operations include push and pop operations (enqueue and dequeue) (pg. 498-499). However, STONE does not teach that the operations have opposing end variants of each. Official Notice is taken in that it is well known in the art that a deque implemented as an array structure is concurrent accessed by opposing end variants of enqueue and dequeue operations and therefore would be obvious in combination with STONE in order to facilitate concurrent access to both ends of the structure.

6. Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over "a Nonblocking Algorithm for Shared Queues Using Compare-and-Swap" by PRAKASH.

As to claim 11, PRAKASH teaches the linked-list of nodes (linked list) (pg. 549), Official Notice is taken in that it is well known in the art that a singly linked list converts to a doubly linked list by storing a previous pointer in each node. Therefore, it would be obvious to one skilled in art to combine the teachings of PRAKASH with the well-known teaching in order to traverse a linked list in different directions.

Response to Arguments

7. Applicant's arguments filed 1/3/05 have been fully considered but they are not persuasive. Applicant argued that Stone does not anticipate the claims because the queue is non-blocking. The examiner disagrees. Stone states throughout the document that the algorithm allows for concurrent operations and has the following

properties: Every modification of the shared head or tail is expressed as a transition in the psuedocode; The state diagram is closed under the transitions; Queue operations always complete; and Items are dequeued in the order in which they are enqueued. The part of the document Applicant refers to states that the queue algorithm fails and therefore is nonblocking if the enqueuer fails to complete, i.e. processor failure. Since the claims do not address processor failure or any situations of processor failure, the queue algorithm of Stone meets the limitations of the claims as disclosed because Stone explicitly states that the algorithm is nondelaying: it is never necessary for one processor to wait for another processor to do something (pg. 503), therefore non-blocking.

Applicant argued that Prakash does not teach the limitation "wherein at least for those of the valid states other than the one or more boundary states, opposing-end ones of the access operations are disjoint" because the Enqueue procedure includes a state entitled "cooperate in dequeuing the object". The examiner disagrees. Prakash explicitly states that "Enqueuing and dequeuing may proceed in parallel, however, except for the case in which the head is the same as the tail. Here, the enqueue and dequeue operations are serialized (pg. 551, D. Proof of Correctness). Applicant appears to be referring to the figure of the Enqueue Procedure. However, the Cooperate in dequeuing the object has no program code in the figure. Figure 5 has comments, program code, and explanations throughout. The Italicize text that Applicant refers to is not program code. Therefore the algorithms nevers helps the dequeue operations or the extension of the helping such that the algorithms are not disjoint

cannot be determined. However, Prakash does explicitly state that the operations do proceed in parallel in cases except when the head and tail are the same. Therefore, the examiner believes Prakash teaches the limitation as disclosed.

Applicant argued that Stone does not teach the means for coordinating competing pop operations by employing an atomic operation to disambiguate a retry state and a boundary condition state. The examiner disagrees. Stone teaches the pop operation (dequeue operation) having an atomic operation (CSDBL) to disambiguate a retry state (whether CC is set to try or false because the algorithm is repeated until the operation is true) and a boundary condition state (determine if next is zero or not zero) based on then-current contents of the end identifier stores (the copy of the head or the copy of the tail) and an element of the concurrent shared object (the count of the object) (see figure 3). Stone also states that new enqueue and dequeue operations can begin at any time and will proceed to successful conclusion as well as the algorithm permits the queue to have concurrent operations. Since dequeue is an operation allowed on the queue, the algorithm of Stone inherently handles multiple dequeue operations. Therefore, Stone teaches the coordinating of multiple dequeue operations.

Applicant argues that Stone does not suggest mediating concurrent non-blocking access "by performing during execution of each of the access operations, an atomic update of a respective one of the opposing end indices and of an array element corresponding thereto." The examiner disagrees. Stones enqueue and dequeue algorithms not only change a pointer but also increment a count for the node. The enqueue operations as a atomic update operation (CSDBL) that updates a respective

end indices (tail) and increments a count stored in the element. Therefore, Stone teaches the limitation as disclosed.

Conclusion

8. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Lewis A. Bullock, Jr. whose telephone number is (571) 272-3759. The examiner can normally be reached on Monday-Friday, 8:30 - 5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

April 15, 2005

LEWIS A. BULLOCK, JR.
PRIMARY EXAMINER